

Re: Metafile EXTTEXTOUT record structure?

Source:

<http://www.tech-archive.net/Archive/Development/microsoft.public.win32.programmer.gdi/2005-10/msg00265.html>

- *From:* "David Thielen" <thielen@xxxxxxxxxxxxxx>
 - *Date:* Fri, 28 Oct 2005 09:01:03 -0700
-

Hi;

Our product may straddle the java/Windows world in a weirder way than any other. We have a reporting package, <http://www.windwardreports.com> that is a pure java solution. And we have customers running it on solaris, linux, AIX, AS/400, some mainframes, as well as Windows. So the engine must be pure java.

The report templates are created in Word. So we have to parse rtf and xml (WordML) files – totally in java. The wmf/emf comes in to play in three places.

First, all images in a rtf file have a \nonship copy of the image that is a wmz file. And WordPad needs that part to display an image. So we have to create wmz images for any bitmap be it bmp, jpg, png, etc.

Second, a wmf or emf can be inserted in a document. But the final report could be in the html format. So in that case we have to convert the wmf/emf into png for the output. And pdf can handle wmf, but not emf or wmz/emz so again we need to parse the metafile in that case.

Third, when there is an embedded OLE object, a picture of the object output is stored in a wmf (rtf)/wmz (xml) format. This allows us to display the OLE object in the output document as a png image.

Does it all make sense now?

—

thanks – dave

"Vipin[MVP]" wrote:

- > I am a bit curious. So if I understand, you are trying to do this from java,
- > that would mean
- > you want the same code to work on multiple platforms. If your target is just
- > windows, why don't you spawn out a helper process
- > in c/c++ which can make use of win32 GDI apis and then you can always use
- > IPC to communicate what you want.

Re: Metafile EXTTEXTOUT record structure?

> Isn't that the easier approach?
>
> --
> Vipin Aravind
> MVP [Windows – Printing/Imaging]
>
> "David Thielen" <thielen@xxxxxxxxxxxxxx> wrote in message
> news:2A3FDF41-1E4E-44D6-9EBB-C8B181CEB2B7@xxxxxxxxxxxxxxxxxxxxx
>> Hello;
>>
>> Yes, but I need to know what the order is in memory too. I am accessing
>> this
>> file from java so I cannot use the GDI calls to access it. And some of the
>> code samples I have seen show that sometimes there is a clip rectangle in
>> the
>> middle of the structure – so I need to know when and where that is placed.
>>
>> --
>> thanks – dave
>>
>>
>> ""Terry Fei[MSFT]"" wrote:
>>
>>> Hi Dave,
>>>
>>> I viewed the record of EXTTEXTOUT in the Debugger and found the
>>> parameter's
>>> order in memory doesn't match the order of ExtTextOut API parameters. In
>>> a
>>> word, the order of parameters isn't 1:1 from record data in memory to
>>> ExtTextOut API. Although the order in memory doesn't match ExtTextOut's,
>>> the data in memory is sufficient to make ExtTextOut invoked successfully
>>> (GDI know how to arrange parameters for ExtTextOut API from memory).
>>>
>>> If you have any questions or concerns, please feel free to let me know.
>>> Thanks! :)
>>>
>>> Best Regards,
>>>
>>> Terry Fei[MSFT]
>>> Microsoft Community Support
>>> Get Secure! www.microsoft.com/security
>>> (This posting is provided "AS IS", with no warranties, and confers no
>>> rights.)
>>>
>>> -----
>>> >Thread-Topic: Metafile EXTTEXTOUT record structure?
>>> >thread-index: AcXa+vUbCPUH6nSGRfSfcHda6r07Jw==
>>> >X-WBNR-Posting-Host: 199.45.247.98
>>> >From: "=?Utf-8?B?RGF2aWQgVGhpZWxlbg==?" <thielen@xxxxxxxxxxxxxx>

Re: Metafile EXTTEXTOUT record structure?

>>> >References: <D340F4AC-F942-4C00-8E26-F2C93454F8CB@xxxxxxxxxxxxxx>
>>> <cpMDGbt2FHA.1144@xxxxxxxxxxxxxxxxxxxxxx>
>>> >Subject: RE: Metafile EXTTEXTOUT record structure?
>>> >Date: Thu, 27 Oct 2005 06:33:03 -0700
>>> >Lines: 101
>>> >Message-ID: <9DFB31A2-8BE7-452C-966A-4F349382C077@xxxxxxxxxxxxxx>
>>> >MIME-Version: 1.0
>>> >Content-Type: text/plain;
>>> > charset="Utf-8"
>>> >Content-Transfer-Encoding: 7bit
>>> >X-Newsreader: Microsoft CDO for Windows 2000
>>> >Content-Class: urn:content-classes:message
>>> >Importance: normal
>>> >Priority: normal
>>> >X-MimeOLE: Produced By Microsoft MimeOLE V6.00.3790.0
>>> >Newsgroups: microsoft.public.win32.programmer.gdi
>>> >NNTP-Posting-Host: TK2MSFTNGXA03.phx.gbl 10.40.2.250
>>> >Path: TK2MSFTNGXA01.phx.gbl!TK2MSFTNGXA03.phx.gbl
>>> >Xref: TK2MSFTNGXA01.phx.gbl microsoft.public.win32.programmer.gdi:7216
>>> >X-Tomcat-NG: microsoft.public.win32.programmer.gdi
>>> >
>>> >Hello;
>>> >
>>> >My question is not is there all data needed in those structures, it's
>>> >that
>>> >the data in a WMF for an EXTTEXTOUT doesn't match the structures. For an
>>> >EXTTEXTOUT the data comes as:
>>> >
>>> >y //word
>>> >x //word
>>> >count //word
>>> >flags //word
>>> >//optional rect
>>> >test //byte[]
>>> >
>>> >The above is based on code in two open source wmf readers as well as
>>> >viewing
>>> >this record in my code in the debugger. But this does not match
>>> >EMREXTTEXTOUTA or EMRTEXT.
>>> >
>>> >So what is going on here? I don't think I am wrong on the struct above
>>> >as
>>> >it
>>> >is giving me the values in the metafile - but it doesn't match the
>>> >documentation.
>>> >
>>> >--
>>> >thanks - dave
>>> >
>>> >
>>> >""Terry Fei[MSFT]"" wrote:

Re: Metafile EXTTEXTOUT record structure?

```
>>>>
>>>> Hi Dave,
>>>> Welcome to NewsGroup!
>>>>
>>>> Based on my understanding, you think that EMREXTTEXTOUT structure is
>>>> not
>>>> complete and lacks some necessary information (for example:bounding
>>>> rectangle) . If I misunderstood you, please feel free to let know.
>>>> The EMREXTTEXTOUTA and EMREXTTEXTOUTW structures contain members for
>>>> the
>>>> ExtTextOut, TextOut, or DrawText enhanced metafile records. If we
>>>> looked
>>>> into this structure, we will find it provides sufficient information
>>>> to
>>>> make metafile drawn. I will list all relevant member of this structure
>>>> as
>>>> follows,
>>>>
>>>> typedef struct tagEMREXTTEXTOUTA
>>>> {
>>>> EMR emr;
>>>> RECTL rclBounds; // Inclusive–inclusive bounds in
>>>> device
>>>> units
>>>> DWORD iGraphicsMode; // Current graphics mode
>>>> FLOAT exScale; // X and Y scales from Page units to
>>>> .01mm
>>>> units
>>>> FLOAT eyScale; // if graphics mode is
>>>> GM_COMPATIBLE.
>>>> EMRTEXT emrtext; // This is followed by the string and
>>>> spacing
>>>> // array
>>>> } EMREXTTEXTOUTA, *PEMREXTTEXTOUTA,
>>>> EMREXTTEXTOUTW, *PEMREXTTEXTOUTW;
>>>>
>>>> // Base record type for the enhanced metafile.
>>>> typedef struct tagEMR
>>>> {
>>>> DWORD iType; // Enhanced metafile record type
>>>> DWORD nSize; // Length of the record in bytes.
>>>> // This must be a multiple of 4.
>>>> } EMR, *PEMR;
>>>>
>>>> typedef struct _RECTL /* rcl */
>>>> {
>>>> LONG left;
>>>> LONG top;
>>>> LONG right;
>>>> LONG bottom;
>>>> } RECTL, *PRECTL, *LPRECTL;
```

Re: Metafile EXTTEXTOUT record structure?

Re: Metafile EXTTEXTOUT record structure?

◆ **RE: Metafile EXTTEXTOUT record structure?**

◇ From: "Terry Fei[MSFT]"

◆ **RE: Metafile EXTTEXTOUT record structure?**

◇ From: David Thielen

- Prev by Date: **Re: Listing of known metafile ESCAPE commands?**
- Next by Date: **Re: EMF – coordinate transforms**
- Previous by thread: **RE: Metafile EXTTEXTOUT record structure?**
- Next by thread: **Re: Metafile EXTTEXTOUT record structure?**
- Index(es):
 - ◆ **Date**
 - ◆ **Thread**